

It is difficult to tell if there is a Condorcet spanning tree

Andreas Darmann¹

Received: 16 September 2015 / Accepted: 22 January 2016 / Published online: 6 February 2016
© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract We apply the well-known Condorcet criterion from voting theory outside of its classical framework and link it with spanning trees of an undirected graph. In situations in which a network, represented by a spanning tree of an undirected graph, needs to be installed, decision-makers typically do not agree on the network to be implemented. Instead, each of these decision-makers has her own ideal conception of the network. In order to derive a group decision, i.e., a single spanning tree for the entire group of decision-makers, the goal would be a spanning tree that beats each other spanning tree in a simple majority comparison. When comparing two dedicated spanning trees, a decision-maker will be considered to be more satisfied with the one that is “closer” to her proposal. In this context, the most basic and natural measure of distance is the usual set difference: we simply count the number of edges the spanning tree has in common with the proposal of the decision-maker. In this work, we show that it is computationally intractable to decide (1) if such a spanning tree exists, and (2) if a given spanning tree satisfies the Condorcet criterion.

Keywords Spanning tree · Condorcet · Computational complexity · Social choice theory

1 Introduction

Dating back more than two centuries ([Condorcet 1785](#)), the Condorcet criterion is one of the fundamental and most important concepts in voting theory (see [Brams and Fishburn 2002](#) for a survey). Originally formulated for single winner elections, the

✉ Andreas Darmann
andreas.darmann@uni-graz.at

¹ Institute of Public Economics, University of Graz, Graz, Austria

criterion states that, if an alternative exists that wins¹ by a simple majority against each other alternative in a pairwise comparison, that alternative should be declared the winner of the election. The Condorcet criterion has been generalized to the election of committees in several ways, e.g., by Fishburn (1981), Gehrlein (1985), Kaymak and Sanver (2003), and more recently by Elkind et al. (2014) and Darmann (2013a).

In the last decades, researchers have started to use the Condorcet criterion outside of its classical framework, often under the label of *popularity*. In particular, this has been done in the context of a popular matching in a bipartite graph, i.e., a matching which beats each other matching in the graph in a simple majority comparison (see, e.g., Gärdenfors 1975). In that respect, a lot of works are concerned with the computational complexity involved in finding such a popular matching, examples being the works of Abraham et al. (2005), Biró et al. (2010), and Huang and Kavitha (2011). In another recent work, Zuylen et al. (2014) consider the problem of finding a popular ranking from a computational viewpoint. In their framework, given a set of alternatives, each member of a set of agents proposes her ideal solution, i.e., individual ranking of the alternatives. The goal is to find a ranking π meeting the property that there is no other ranking π' which beats π in a simple majority comparison. In Zuylen et al. (2014), these comparisons are performed by measuring the distance of the considered ranking to each of the agents' rankings by means of the Kendall-tau distance.

We apply the Condorcet criterion to spanning trees of an undirected graph. Finding spanning trees in an undirected graph is a central task in operations research, with numerous applications in the construction of physical networks such as telecommunication networks, power transmission lines, and pipeline or highway systems (see also Ahuja et al. 1993). In this paper, we consider the situation in which such a network needs to be installed. Typically, however, the corresponding decision-makers involved do not agree on the network. Instead, each of them decides on her own ideal solution, i.e., spanning tree, which she proposes. Clearly, each of the decision-makers would like to see her ideal spanning tree being implemented, which is impossible except in the trivial case of all decision-makers proposing the same spanning tree. Now, it is natural to assume that, when considering two dedicated spanning trees, each of the decision-makers will be more happy with the one that is "closer" to her proposal. The most basic and natural measure of distance in this context is to simply count the number of edges the spanning tree has in common with the proposal of the decision-maker, i.e., to consider the usual set difference. Finally, to decide on an overall solution for the whole group of decision-makers, the well-studied Condorcet criterion is a meaningful and plausible criterion.

As an illustrative example we consider the member states of the European Union deciding on the layout of a communication network (e.g., a high security connection of law enforcement agencies) or a crucial supply network such as a gas or oil pipeline. These networks are generally designed as spanning trees connecting all countries represented by vertices in a graph. The decision-makers could be the countries themselves, or board members of some EU institution. For political and historical reasons involving trust, self-interest, or stability concerns, different decision-makers have different

¹ Respectively, in its weaker version, does not lose by a simple majority against any other alternative in a pairwise comparison

perceptions which links (i.e., edges) between the countries should be used in order to establish the whole network. In particular, the decision-makers could have different conceptions of the entire network because of political relations and alliances (especially in case the decision-makers are the countries themselves), or due to concerns regarding the security of supply and political stability. In addition, different decision-makers could base their conceptions on different criteria, which typically results in differing networks. Finding a plausible solution which takes into account the interests of all the decision-makers in such a setting is clearly an important aggregation task.

In principle, we are thus concerned with the aggregation of spanning trees of an undirected graph into a single spanning tree. Hence, this work is somewhat related to the axiomatically motivated work of [Endriss and Grandi \(2012\)](#), where, in a preference-based environment, each member of a set of agents proposes a directed graph, which need to be aggregated into a single graph. More formally, similar to the work of [Zuylen et al. \(2014\)](#), in our framework each member of a set of agents proposes her favourite solution, i.e., spanning tree of the graph. Now, a (weak) Condorcet tree is a spanning tree T such that for each other spanning tree T' , T beats (is not beaten by) T' in a pairwise majority comparison. In order to perform these comparisons, we measure the distance between two spanning trees by means of the set difference. In this work, we consider two natural decision problems in connection with a (weak) Condorcet tree from a computational perspective. In the first one, we consider the question if such a spanning tree exists. In the second problem, we ask if a given specific spanning tree is a (weak) Condorcet tree.

The first question is related to the work of [Darmann \(2014\)](#), where instead of applying the Condorcet criterion, the goal is to find a spanning tree which does not deviate from each of the individual spanning trees by more than a given number of edges. Note that that problem is structurally rather different from the ones considered in this paper. For instance, a (weak) Condorcet tree might be identical with a particular spanning tree proposed by many agents while it might be completely disjoint from some other proposed spanning trees. [Darmann \(2014\)](#), however, proves that it is NP-complete to decide if such a tree exists. In this work, by providing coNP-hardness results we show that deciding if a (weak) Condorcet tree exists is also a computationally hard problem.

The second question is in the spirit of the seminal paper by [Bartholdi et al. \(1989\)](#), who show that both for the Dodgson and Kemeny voting rule it is NP-hard to decide if a given candidate is a winner of an election. The related work of [Darmann \(2013b\)](#) focuses on the second question, in a different framework and using the terminology of popularity. In particular, [Darmann \(2013b\)](#) provides computational complexity results for the problem of deciding if a given spanning tree is popular² with respect to scoring functions adopted from voting rules, with the focus on approval voting and Borda voting. The approval voting scenario of [Darmann \(2013b\)](#) is closely related to our approach, the main difference being that instead of whole spanning trees in [Darmann \(2013b\)](#) the agents propose single edges (which do not induce a certain structure) that should be included in the spanning tree. From [Darmann \(2013b\)](#) it follows that

² In this context, “popular tree” translates into “weak Condorcet tree” in our setting.

deciding if a given spanning tree is a weak Condorcet tree is **coNP**-complete already if each of the agents propose two edges only. The computational complexity of the problem when the proposed edges form a certain structure, in particular a spanning tree of the given graph, is not investigated in Darmann (2013b) and, to the best of our knowledge, has not been studied before. In this work, we add to the result of Darmann (2013b) by showing that deciding if a given spanning tree is a (weak) Condorcet tree is **coNP**-complete if the agents propose a whole solution, i.e., spanning tree, instead of single edges.³

2 Problem definition

We briefly introduce the formal framework of this paper. An undirected graph $G = (V, E)$ consists of a finite set V and a set E of two-element subsets of V . We call the elements of V vertices and the elements of E edges. A subset $T \subseteq E$ with $|E| = |V| - 1$ is called a *spanning tree* of G , if (V, T) contains no cycle. Note that for any two spanning trees T_1, T_2 of G we have $|T_1 \setminus T_2| = |T_2 \setminus T_1|$.

Let $G = (V, E)$ be an undirected graph and let A be a set of agents. For each agent $a \in A$, we are given a spanning tree T_a of G . Then, a spanning tree T of G is called a *Condorcet tree* if for all spanning trees \tilde{T} of G , $\tilde{T} \neq T$,

$$\#\{a \in A : |T \setminus T_a| < |\tilde{T} \setminus T_a|\} > \#\{a \in A : |\tilde{T} \setminus T_a| < |T \setminus T_a|\}$$

holds. Similarly, a spanning tree T of G is called a *weak Condorcet tree*, if for all spanning trees \tilde{T} of G , $\tilde{T} \neq T$,

$$\#\{a \in A : |T \setminus T_a| < |\tilde{T} \setminus T_a|\} \geq \#\{a \in A : |\tilde{T} \setminus T_a| < |T \setminus T_a|\}$$

is satisfied. Clearly, a Condorcet tree is also a weak Condorcet tree, but the reverse does not necessarily hold.

In the first decision problem considered in this paper we ask for the existence of a Condorcet tree.

Definition 1 (CONDORCET-TREE)

GIVEN:	Set A of agents, undirected graph $G = (V, E)$, spanning trees T_a of G for $a \in A$.
QUESTION:	Is there a Condorcet tree?

Analogously, **WEAK-CONDORCET-TREE** is the problem of deciding if there exists a weak Condorcet tree.

The second type of decision problem we consider is to decide if a given spanning tree is a (weak) Condorcet tree.

³ From a technical viewpoint, note that the main proofs in Darmann (2013b, 2014) reduce from variants of 2-SAT, while in our main proof construction we provide a reduction from 3-UNSAT.

Definition 2 (CONDORCET-TREE WINNER)

GIVEN:	Set A of agents, undirected graph $G = (V, E)$, spanning trees T_a of G for $a \in A$, distinguished spanning tree T of G .
QUESTION:	Is T a Condorcet tree?

Analogously, in WEAK-CONDORCET-TREE WINNER we ask if a given spanning tree is a weak Condorcet tree.

Let $\omega : E \rightarrow \mathbb{N}_0$ be defined by $\omega(e) := |\{a \in A : e \in T_a\}|$. As captured by the lemma below, the candidates for a weak Condorcet tree – and hence for a Condorcet tree – can be restricted to the set of maximum spanning trees in G with respect to ω , i.e., spanning trees in G that have the maximum weight with respect to ω .

Lemma 1 *If T is a weak Condorcet tree, then T is a maximum spanning tree with respect to ω .*

Proof Assume that in the considered graph $G = (V, E)$, T is not a maximum spanning tree w.r.t. ω . Then, by the path optimality condition for maximum spanning trees (cf. Ahuja et al. 1993) there must be an edge $e' = \{u, v\} \in E \setminus T$ such that $\omega(e') > \omega(e'')$ for some edge e'' contained in the unique path in T which connects the nodes u and v . Clearly, $T' := T \cup \{e'\} \setminus \{e''\}$ is again a spanning tree of G . Let $A_{T'} = \{a \in A : |T' \setminus T_a| < |T \setminus T_a|\}$ and $A_T = \{a \in A : |T \setminus T_a| < |T' \setminus T_a|\}$. Comparing T' to T yields

$$|A_{T'}| - |A_T| = \omega(e') - \omega(e'') > 0.$$

Hence, T is not a weak Condorcet tree. □

Lemma 1 is interesting in its own right as similar results have been stated for multiple referenda (Laffond and Lainé 2009) and committee selection (Brams et al. 2007a,b). In particular, Laffond and Lainé (2009) generalize a result of Brams et al. (2007a,b) which states that, using approval ballots and alternative-wise majority voting, the winning committee maximizes the total sum of the voters' utilities when individual utility is measured by the number of alternatives included in the committee.

In what follows, we show that each of the problems CONDORCET-TREE, WEAK-CONDORCET-TREE, CONDORCET-TREE WINNER, and WEAK-CONDORCET-TREE WINNER is computationally hard.

3 Computational intractability results

We start by showing that CONDORCET-TREE is computationally intractable.

Theorem 1 CONDORCET-TREE is coNP-hard.

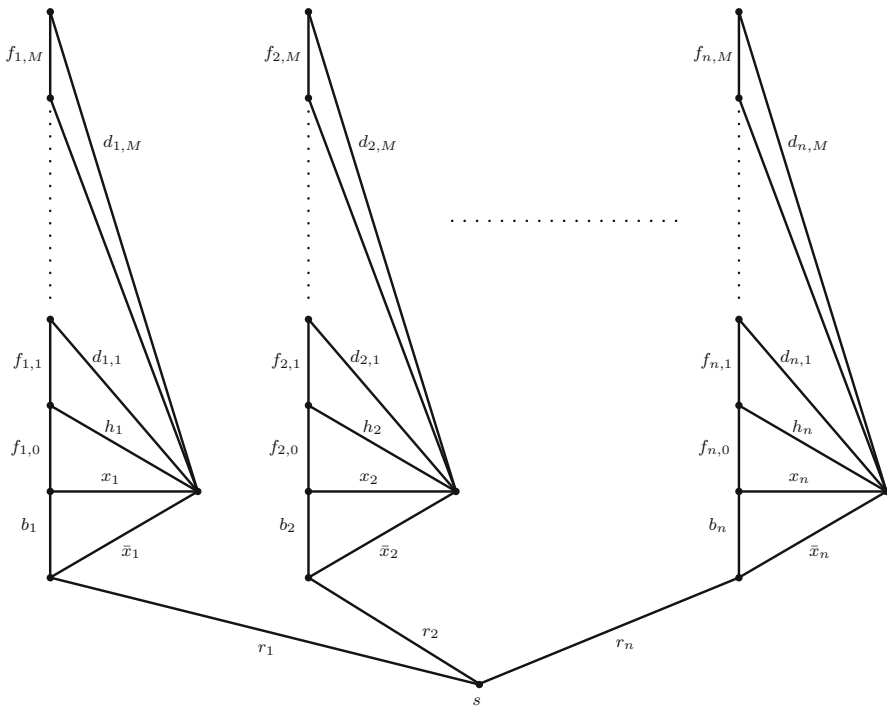


Fig. 1 Graph G in instance \mathcal{V} of CONDORCET-TREE

Proof We provide a reduction from the **coNP**-complete problem 3-UNSAT, the complement of the classical **NP**-complete problem 3-SAT. In an instance of 3-UNSAT, we are given a set X of variables and a set C of (disjunctive) clauses made up of exactly 3 literals (a literal is a variable or the negation of a variable in X); the task is to decide if C is unsatisfiable, i.e., if there is no truth assignment for X that satisfies all the clauses in C . 3-UNSAT remains **coNP**-complete if the number of occurrences of each variable is bounded by 4 (3-SAT is known to be **NP**-complete under these restrictions (Tovey 1984); obviously, 3-UNSAT is **coNP**-complete under the same restrictions).

Let $\mathcal{U} = (X, C)$ be such an instance of 3-UNSAT, with $X = \{x_1, \dots, x_n\}$ and $C = \{C_1, \dots, C_m\}$ for some $n, m \in \mathbb{N}$. W.l.o.g. we assume that a clause does not contain both x_t and \bar{x}_t for some $t \in \{1, \dots, n\}$, since such a clause is satisfied by any feasible truth assignment. Let Z be the set of literals, i.e., $Z = \{x_t, \bar{x}_t \mid 1 \leq t \leq n\}$. For $1 \leq j \leq m$, we write $C_j = (z_{j_1} \vee z_{j_2} \vee z_{j_3})$ with $z_{j_p} \in Z$, $p \in \{1, 2, 3\}$. We identify a truth assignment ϕ with the set of literals set true under ϕ .

Step I: Constructing instance \mathcal{V} of CONDORCET-TREE. From \mathcal{U} , we construct an instance \mathcal{V} of CONDORCET-TREE as follows.

Step Ia: Defining the graph. Let $M := m + 2n^2 + 5n + 1$. First, in order to define a graph $G = (V, E)$ (see Fig. 1), we set

$$V := \{s\} \cup \{s_t, u_t, v_t, w_t \mid 1 \leq t \leq n\} \cup \{y_{t,q} \mid 1 \leq t \leq n, 1 \leq q \leq M\}.$$

In addition, for $1 \leq t \leq n$, we define the edges $r_t = \{s, s_t\}$, $b_t = \{s_t, u_t\}$, $x_t = \{u_t, w_t\}$, $\bar{x}_t = \{s_t, w_t\}$, $h_t = \{v_t, w_t\}$, and $f_{t,0} = \{u_t, v_t\}$. Finally, for $1 \leq t \leq n$, $1 \leq q \leq M$, we define the edges $d_{t,q} = \{y_{t,q}, w_t\}$ and $f_{t,q} = \{y_{t,q-1}, y_{t,q}\}$, where $y_{t,0} := v_t$. Let $R := \bigcup \{r_t\}$, $B := \bigcup \{b_t\}$, $H := \bigcup \{h_t\}$, $D := \bigcup \{d_{t,q}\}$, and $F := \bigcup \{f_{t,q}\}$, where the indices in the set unions run over all possible values of t and q respectively. Identifying a literal with an edge of the same label, we formally define

$$E := R \cup B \cup D \cup F \cup H \cup Z.$$

Note that $|V| = 1 + 4n + nM$ and $|E| = 6n + 2nM$. Hence, any spanning tree \tilde{T} of G consists of

$$|\tilde{T}| = |V| - 1 = n(M + 4) \quad (1)$$

edges.

Step Ib: Introducing the agents. We introduce a set $A = A_H \cup A_D \cup A_C \cup A_Z$ of agents.

The agent set A_H consists of $m + 2n^2 + n$ agents, each $h \in A_H$ proposing the same spanning tree

$$T_h = T = R \cup B \cup F \cup H.$$

For $1 \leq q \leq M$, let $D_q := \{d_{t,q} | 1 \leq t \leq n\}$. The next set of agents $A_C = \{\gamma_j | 1 \leq j \leq m\}$ is such that agent γ_j , in order to represent clause C_j , proposes the tree

$$T_{\gamma_j} := R \cup B \cup F \cup \{z_{j1}, z_{j2}, z_{j3}\} \cup (D_j \setminus \{d_{j1,j}, d_{j2,j}, d_{j3,j}\}).$$

Next, we introduce the $2n^2 + n$ agents in A_Z , where, for $1 \leq t \leq n$ and $1 \leq q \leq 2n + 1$, agent $\zeta_{t,q}$ proposes the spanning tree

$$T_{\zeta_{t,q}} := R \cup F \cup \{x_t, \bar{x}_t\} \cup (B \setminus \{b_t\}) \cup (D_{m+(t-1) \cdot (2n+1)+q} \setminus \{d_{t,m+(t-1) \cdot (2n+1)+q}\}).$$

Finally, the agents in $A_D = \{\delta_i | 1 \leq i \leq 4n + 1\}$ are such that δ_i proposes the spanning tree

$$T_{\delta_i} = R \cup B \cup F \cup D_{m+2n^2+n+i}.$$

It is not hard to verify that each proposed solution is indeed a spanning tree of G . Note that

$$|A| = (m + 2n^2 + n) + m + (2n^2 + n) + (4n + 1) = 2m + 4n^2 + 6n + 1 \quad (2)$$

holds. In addition, observe that

- each edge in H is contained in the spanning tree T proposed by $m + 2n^2 + n$ agents,
- the set $R \cup F$ is contained in each of the proposed spanning trees,
- each edge in B is contained in all but $2n + 1$ of the proposed spanning trees, while

- each edge in D is contained in at most one proposed spanning tree and
- each edge in Z is contained in at most $2n + 5$ of the proposed spanning trees.

From these observations, it is easy to conclude that T is the unique maximum spanning tree w.r.t. the edge weights $\omega(e)$, i.e., the only candidate for a Condorcet tree (see Lemma 1). Note that

- for each $h \in A_H$ we have $|T \setminus T_h| = 0$,
- for each $\gamma \in A_C$ it holds that $|T \setminus T_\gamma| = |H| = n$,
- for each $\zeta_{t,q} \in A_Z$ we have $|T \setminus T_{\zeta_{t,q}}| = |(B \cup H) \setminus (B \setminus \{b_t\})| = |H \cup \{b_t\}| = n + 1$,
and
- for each $\delta \in A_D$, $|T \setminus T_\delta| = |H| = n$ holds.

Step II: Proving the reduction. We complete the proof by showing that the following claim holds:

\mathcal{U} is a “yes”-instance of 3-UNSAT if and only if \mathcal{V} is a “yes”-instance of CONDORCET-TREE.

“ \Leftarrow ”: Let \mathcal{V} be a “yes”-instance of CONDORCET-TREE. As a consequence, T is the unique Condorcet tree. Assume that \mathcal{U} is a “no”-instance of 3-UNSAT, i.e., there is a truth assignment ϕ that satisfies all clauses in C . We show that this leads to the conclusion that T is not a Condorcet tree in \mathcal{V} and hence to a contradiction.

Recall that $|\phi| = n$ and consider the spanning tree

$$T' := \phi \cup R \cup F \cup B.$$

We can derive the following conclusions concerning the agents in A :

- agents in A_C : for each γ_j we have $|T' \setminus T_{\gamma_j}| = |\phi \setminus \{z_{j_1}, z_{j_2}, z_{j_3}\}| \leq n - 1$, since the satisfying truth assignment ϕ contains at least one literal of the clause C_j . Thus, for each $\gamma \in A_C$ we have $|T' \setminus T_\gamma| < |T \setminus T_\gamma|$.
- agents in A_Z : for each $1 \leq t \leq n$, we get

$$\begin{aligned} |T' \setminus T_{\zeta_{t,q}}| &= |(\phi \cup B) \setminus (\{x_t, \bar{x}_t\} \cup B \setminus \{b_t\})| \\ &= |\{b_t\} \cup (\phi \setminus \{x_t, \bar{x}_t\})| \\ &= 1 + (n - 1) \\ &= n \end{aligned}$$

because per definition ϕ contains exactly one of $\{x_t, \bar{x}_t\}$ for each t . Hence, for each $\zeta \in A_Z$ we have $|T' \setminus T_\zeta| < |T \setminus T_\zeta|$.

- agents in A_D : for each $\delta \in A_D$, $|T' \setminus T_\delta| = |\phi| = n$ holds. Thus, for each $\delta \in A_D$ we have $|T' \setminus T_\delta| = |T \setminus T_\delta|$.

On the other hand, the agents in A_H propose the spanning tree T . Hence, we get

$$\#\{a \in A : |T \setminus T_a| < |T' \setminus T_a|\} = |A_H| = m + 2n^2 + n$$

and

$$\#\{a \in A : |T' \setminus T_a| < |T \setminus T_a|\} = |A_C| + |A_Z| = m + 2n^2 + n.$$

Thus, T is not a Condorcet tree. Hence there is no Condorcet tree in \mathcal{V} , which contradicts our assumption. As a consequence, \mathcal{U} must be a “yes”-instance of 3-UNSAT.

“ \Rightarrow ”: Let \mathcal{V} be a “no”-instance of CONDORCET-TREE. Thus, T is not a Condorcet tree, i.e., there must be a spanning tree $\hat{T} \neq T$ such that

$$\#\{a \in A : |T \setminus T_a| < |\hat{T} \setminus T_a|\} \leq \#\{a \in A : |\hat{T} \setminus T_a| < |T \setminus T_a|\} \quad (3)$$

holds. Clearly, all $a \in A_H$ have $|T \setminus T_a| < |\hat{T} \setminus T_a|$. Hence, (3) implies

$$\#\{a \in A : |\hat{T} \setminus T_a| < |T \setminus T_a|\} \geq |A_H|. \quad (4)$$

The proof proceeds in four steps.

(i) *First, we show that we can assume that $\hat{T} \cap H = \emptyset$.* Assume $\hat{T} \cap H \neq \emptyset$. Create the spanning tree T' of G from \hat{T} by replacing each edge $h_i \in H$ by an edge $e \in \{f_{i,0}, x_i\}$ which (a) is not included in \hat{T} and (b) satisfies that $\hat{T} \cup \{e\} \setminus \{h_i\}$ is in fact a spanning tree of G . Obviously, $|T \setminus T_a| < |T' \setminus T_a|$ and $|T \setminus T_a| < |\hat{T} \setminus T_a|$ hold for $a \in A_H$. However, since there is no tree proposed by an agent in $A \setminus A_H$ which contains an edge in H , we have

$$|T' \setminus T_a| \leq |\hat{T} \setminus T_a|$$

for $a \in A \setminus A_H$. Hence,

$$\#\{a \in A : |T \setminus T_a| < |T' \setminus T_a|\} \leq \#\{a \in A : |T' \setminus T_a| < |T \setminus T_a|\}$$

holds as well.

(ii) *Second, we show that $(R \cup B \cup F) \subset \hat{T}$ holds.* By construction, $R \subset \hat{T}$. Let $\kappa := |(B \cup F) \setminus \hat{T}|$. Assume that $\kappa \geq 1$ holds.

If $\kappa > n$, then each $\delta \in A_D$ has $|T \setminus T_\delta| = n < \kappa \leq |\hat{T} \setminus T_\delta|$, because $[(B \cup F) \setminus \hat{T}] \subseteq (T_\delta \setminus \hat{T})$ and $|T_\delta \setminus \hat{T}| = |\hat{T} \setminus T_\delta|$ hold. Thus,

$$\begin{aligned} \#\{a \in A : |T \setminus T_a| < |\hat{T} \setminus T_a|\} &\geq |A_H| + |A_D| \\ &= (m + 2n^2 + n) + 4n + 1 \\ &> \frac{|A|}{2} \end{aligned}$$

because the total number of agents is $|A| = 2m + 4n^2 + 6n + 1$ (stated in (2)). This contradicts (3).

Hence, $\kappa \leq n$ must hold. However, \hat{T} can have a non-empty intersection with at most $2n$ from the sets D_{m+2n^2+n+i} , $1 \leq i \leq 4n + 1$, because otherwise \hat{T} contains at least

$$\begin{aligned} |R| + (|B \cup F| - \kappa) + (2n + 1) &= n + (n + n(M + 1) - \kappa) + (2n + 1) \\ &= nM + 5n + 1 - \kappa \\ &\geq nM + 4n + 1 \end{aligned}$$

edges, which contradicts (1). Hence, for at least $2n + 1$ agents $\delta_i \in A_D$ — i.e., for at least $2n + 1$ of the indices $i \in \{1, \dots, 4n + 1\}$ — we have $|\hat{T} \setminus T_{\delta_i}| = |T_{\delta_i} \setminus \hat{T}| = \kappa + |D_{m+2n^2+n+i}| \geq 1 + n$. Since $|T \setminus T_\delta| = n$ holds for each $\delta \in A_D$,

$$\begin{aligned} \#\{a \in A : |T \setminus T_a| < |\hat{T} \setminus T_a|\} &\geq |A_H| + 2n + 1 \\ &= (m + 2n^2 + n) + 2n + 1 \\ &> \frac{|A|}{2} \end{aligned}$$

follows. Again this contradicts (3). Thus, $\kappa = 0$, i.e., $(B \cup F) \subset \hat{T}$ and hence $(R \cup B \cup F) \subset \hat{T}$ holds.

(iii) *Third, we show that $|\{x_t, \bar{x}_t\} \cap \hat{T}| = 1$ holds for each $1 \leq t \leq n$.* From $B \subset \hat{T}$ (see (ii)), it follows that $|\{x_t, \bar{x}_t\} \cap \hat{T}| \leq 1$ holds for each $1 \leq t \leq n$, because otherwise \hat{T} would contain a cycle which contradicts with the fact that \hat{T} is a spanning tree. Assume that for some t , $|\{x_t, \bar{x}_t\} \cap \hat{T}| = 0$ holds. By (ii), Eq. (1) implies that \hat{T} can have a non-empty intersection with at most n from the sets $D_{m+(t-1)(2n+1)+q}$, $1 \leq q \leq 2n + 1$. Thus, for at least $n + 1$ indices $q \in \{1, \dots, 2n + 1\}$ we have

$$\begin{aligned} |T_{\zeta_{t,q}} \setminus \hat{T}| &= |\{x_t, \bar{x}_t\} \cup \{D_{m+(t-1)(2n+1)+q} \setminus \{d_{t,m+(t-1)(2n+1)+q}\}| \\ &= 2 + (n - 1) \\ &= n + 1. \end{aligned}$$

Recalling that $|\hat{T} \setminus T_{\zeta_{t,q}}| = |T_{\zeta_{t,q}} \setminus \hat{T}|$ holds, we hence can conclude that $|\hat{T} \setminus T_{\zeta_{t,q}}| = |T \setminus T_{\zeta_{t,q}}|$ is satisfied for at least $n + 1$ agents in A_Z .

Analogously, \hat{T} can have a non-empty intersection with at most n from the sets D_{m+2n^2+n+i} , $1 \leq i \leq 4n + 1$. Thus, at most n of the voters $\delta \in A_D$ have $|\hat{T} \setminus T_\delta| < n$ and hence $|\hat{T} \setminus T_\delta| < |T \setminus T_\delta|$. As a consequence, we get

$$\#\{a \in A : |T \setminus T_a| < |\hat{T} \setminus T_a|\} \geq |A_H| = m + 2n^2 + n$$

while

$$\begin{aligned} \#\{a \in A : |\hat{T} \setminus T_a| < |T \setminus T_a|\} &\leq |A_C| + (|A_Z| - (n + 1)) + n \\ &= m + (2n^2 + n - n - 1) + n \\ &= m + 2n^2 + n - 1. \end{aligned}$$

Therewith,

$$\#\{a \in A : |T \setminus T_a| < |\hat{T} \setminus T_a|\} > \#\{a \in A : |\hat{T} \setminus T_a| < |T \setminus T_a|\}$$

holds, which contradicts (3).

(iv) *Finally, from \hat{T} we derive a truth assignment ϕ that satisfies all clauses in C .* Let $\phi := Z \cap \hat{T}$. From (iii), we know that for each $1 \leq t \leq n$, exactly one of $\{x_t, \bar{x}_t\}$ is contained in ϕ . In other words, $|\phi| = n$ holds and ϕ is a feasible truth assignment. In addition, (iii) implies that $(D \cup H) \cap \hat{T} = \emptyset$ holds because otherwise (1) would

be violated. Hence, we can write \hat{T} as $\hat{T} = R \cup B \cup F \cup \phi$. Consider the agents in $A \setminus A_C$:

- for each agent $h \in A_H$ we have $|\hat{T} \setminus T_h| = n$, whereas $|T \setminus T_h| = 0$; i.e., $|T \setminus T_h| < |\hat{T} \setminus T_h|$.
- for each agent $\zeta_{t,q} \in A_Z$ we have $|\hat{T} \setminus T_{\zeta_{t,q}}| = |(\phi \setminus \{x_t, \bar{x}_t\}) \cup \{b_t\}| = (n-1)+1 = n$, whereas $|T \setminus T_{\zeta_{t,q}}| = 1+n$; i.e., $|\hat{T} \setminus T_{\zeta_{t,q}}| < |T \setminus T_{\zeta_{t,q}}|$.
- for each agent $\delta \in A_D$ we have $|T \setminus T_\delta| = |\hat{T} \setminus T_\delta| = n$.

As a result, due to $|A_H| = m + 2n^2 + n$ and $|A_Z| = 2n^2 + n$, in order to satisfy (3) each agent $\gamma \in A_C$ must have $|\hat{T} \setminus T_\gamma| < |T \setminus T_\gamma| = n$. Because of $D \cap \hat{T} = \emptyset$, the only possibility to achieve this is that, for each $1 \leq j \leq m$, at least one of the edges $\{z_{j1}, z_{j2}, z_{j3}\}$ is contained in \hat{T} . In other words, for each $1 \leq j \leq m$, at least one of the variables $\{z_{j1}, z_{j2}, z_{j3}\}$ that make up clause C_j must be contained in, i.e., set true under ϕ . Thus, ϕ satisfies all clauses in C , i.e., \mathcal{U} is a “no”-instance of 3-UNSAT. \square

In the following theorem, we state an analogous result for WEAK-CONDORCET-TREE.

Theorem 2 WEAK-CONDORCET-TREE is coNP-hard.

Proof The proof follows from the one of Theorem 1 by removing exactly one agent from the set A_H . \square

We now turn to the problems of deciding whether a given spanning tree is a (weak) Condorcet tree. These problems turn out to be coNP-complete.

Theorem 3 CONDORCET-TREE WINNER is coNP-complete.

Proof In the proof of Theorem 1, there is exactly one candidate for a Condorcet tree in instance \mathcal{V} , i.e., the spanning tree T . Thus, in instance \mathcal{V} the answer to question (Q1) “Is there a Condorcet tree?” is “Yes” if and only if the answer to (Q2) “Is T a Condorcet tree?” is “Yes”. It is hence straightforward that CONDORCET-TREE WINNER is coNP-hard.

Finally, membership in coNP is easy to verify. Let \mathcal{C} be an instance of CONDORCET-TREE WINNER, where we ask if $T_{\mathcal{C}}$ is a Condorcet tree. If \mathcal{C} is a “no”-instance, then any tree T' which satisfies $\#\{a \in A : |T_{\mathcal{C}} \setminus T_a| < |T' \setminus T_a|\} \leq \#\{a \in A : |T' \setminus T_a| < |T_{\mathcal{C}} \setminus T_a|\}$ serves as certificate. \square

Analogously, with (the proof of) Theorem 2 we get the following result.

Theorem 4 WEAK-CONDORCET-TREE WINNER is coNP-complete.

4 Conclusion

We have shown that both deciding if a (weak) Condorcet tree exists and deciding if a given spanning tree is a (weak) Condorcet tree, are computationally difficult problems. While the latter problems turn out to be coNP-complete, our coNP-hardness results for CONDORCET-TREE and WEAK-CONDORCET-TREE only provide lower bounds for their computational complexity. A finer placement of these problems in terms of their computational complexity is an interesting open question.

Acknowledgments Open access funding provided by Austrian Science Fund (FWF). This research was supported by the Austrian Science Fund (FWF): P23724-G11 and by the University of Graz project “Wahl-Auswahl-Entscheidung”.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Abraham D, Irving R, Kavitha T, Mehlhorn K (2005) Popular matchings. In: Proceedings of the sixteenth annual ACM-SIAM symposium on discrete algorithms (SODA '05), Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2005, pp 424–432
- Ahuja R, Magnanti T, Orlin J (1993) Network flows: theory, algorithms, and applications. Prentice Hall, New Jersey
- Bartholdi J, Tovey C, Trick M (1989) Voting schemes for which it can be difficult to tell who won the election. *Soc Choice Welf* 6(2):157–165
- Biró P, Irving R, Manlove D (2010) Popular matchings in the marriage and roommates problems. In: Calamoneri T, Díaz J (eds) CIAC, volume 6078 of LNCS. Springer, pp 97–108
- Brams S, Fishburn P (2002) Voting procedures. In: Arrow K, Sen A, Suzumura K (eds) Handbook of social choice and welfare, vol 1. Elsevier, Amsterdam, pp 173–236
- Brams S, Kilgour M, Sanver R (2007a) A minimax procedure for electing committees. *Public Choice* 132(3–4):401–420
- Brams S, Kilgour M, Sanver R (2007b) A minimax procedure for negotiating multilateral treaties. In: Avenhaus R, Zartman I (eds) Diplomacy games. Springer, Berlin, pp 265–282
- Condorcet J (1785) Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix. Imprimerie royale, Paris
- Darmann A (2013a) How hard is it to tell which is a Condorcet committee? *Math Soc Sci* 66(3):282–292
- Darmann A (2013b) Popular spanning trees. *Int J Found Comput Sci* 24(5):655–677
- Darmann A (2014) It is hard to agree on a spanning tree. *Econ Bull* 34(1):34–40
- Elkind E, Lang J, Saffidine A (2014) Condorcet winning sets. *Soc Choice Welf* 44(3):493–517
- Endriss U, Grandi U (2012) Graph aggregation. In: Proceedings of the 4th international workshop on computational social choice (COMSOC-2012). AGH University of Science and Technology, Kraków
- Fishburn P (1981) An analysis of simple voting systems for electing committees. *SIAM J Appl Math* 41(3):499–502
- Gärdenfors P (1975) Match making: assignments based on bilateral preferences. *Behav Sci* 20(3):166–173
- Gehrlein W (1985) The Condorcet criterion and committee selection. *Math Soc Sci* 10(3):199–209
- Huang C, Kavitha T (2011) Popular matchings in the stable marriage problem. In: Aceto L, Henzinger M, Sgall J (eds) ICALP, volume 6755 of LNCS. Springer, pp 666–677
- Kaymak B, Sanver M (2003) Sets of alternatives as Condorcet winners. *Soc Choice Welf* 20(3):477–494
- Laffond G, Lainé J (2009) Condorcet choice and the Ostrogorski paradox. *Soc Choice Welf* 32(2):317–333
- Tovey C (1984) A simplified NP-complete satisfiability problem. *Discrete Appl Math* 8:85–89
- van Zuylen A, Schalekamp F, Williamson D (2014) Popular ranking. *Discrete Appl Math* 165:312–316